

# Synthesis of Switching Rules for Ensuring Reachability Properties of Sampled Linear Systems

Laurent Fribourg

LSV, ENS Cachan & CNRS, 94230 Cachan, France  
fribourg@lsv.ens-cachan.fr

Bertrand Revol

SATIE, ENS Cachan & CNRS, 94230 Cachan, France  
bertrand.revol@satie.ens-cachan.fr

Romain Soulat

LSV, ENS Cachan & CNRS, 94230 Cachan, France  
soulat@lsv.ens-cachan.fr

We consider here systems with piecewise linear dynamics that are periodically sampled with a given period  $\tau$ . At each sampling time, the mode of the system, i.e., the parameters of the linear dynamics, can be switched, according to a switching rule. Such systems can be modelled as a special form of hybrid automata, called “switched systems”, that are automata with an *infinite* real state space. The problem is to find a switching rule that guarantees the system to still be in a given area  $V$  at the next sampling time, and so on indefinitely. In this paper, we will consider two approaches: the *indirect* one that abstracts the system under the form of a finite discrete event system, and the *direct* one that works on the continuous state space.

Our methods rely on previous works, but we specialize them to a simplified context (linearity, periodic switching instants, absence of control input), which is motivated by the features of a focused case study: a DC-DC boost converter built by electronics laboratory SATIE (ENS Cachan). Our enhanced methods allow us to treat successfully this real-life example.

## 1 Introduction

We are interested here in finding rules for switching the modes of (piecewise) linear systems in order to make the variables of the system stay within the limits of given area  $V$ . The systems that we consider are periodically sampled with a given period  $\tau$ . Between two sampling times, the variables follow a certain system of linear differential equations, corresponding to a *mode* among several other ones. At each sampling time, the mode of the system can be switched. Such systems can be modelled as a special form of hybrid automata, called “switched systems”, that are automata with an *infinite* real state space. The problem that we consider here is to find a switching rule that selects a mode ensuring that the system will still be in  $V$  at the next sampling time, and so on indefinitely.

Note that, here, we do not impose that the systems always lies within  $V$  between two sampling times, only at sampling times: if the system goes out of  $V$  between two sampling times, then, due to continuity reasons and because of the “small” size of  $\tau$ , it will still stay within the close neighborhood of  $V$ , and we assume that such a small deviation is acceptable for the system. This makes the problem simpler than the one considered, e.g. in [2], where the system is forced to always stay within  $V$ .

Note also that the problem here is simpler than the one considered in [12], because, here, only the switching rule has to be determined, since the control input is fixed. (In [12], the dynamics is of the form  $\dot{x}(t) = Ax(t) + Bu(t)$ , where  $u(t)$  is not constant, but an input to be synthesized.)

Finally, our problem is much simplified by the fact that, as in [7], the switching instants can only occur at times of the form  $i\tau$  with  $i \in \mathbb{N}$ .

As noted in [2], there are two approaches for solving this kind of problem:

- the *indirect* approach reduces first the system, via abstraction, into a discrete event system (typically, a finite-state automaton); this is done in, e.g., [7]. One can thus identify cycles in the graph of the abstract system, thus inferring possible patterns of modes that enforces the system to stay forever within  $V$ .

- the *direct* approach works directly on the continuous state space; this is done, e.g., in [2]. One can thus infer a *controllable* subspace  $V'$  of  $V$ , within which the existence of a switching rule allowing to stay forever within  $V'$  is guaranteed (see, e.g., [12, 9]).

Often, in the indirect approach, the switching rule can be computed *off line* (under, e.g., the form of a repeated pattern of modes), while the switching rule has to be computed *on line* in the direct approach.

Our methods basically rely on previous works, but we specialize them to the simplified context (linearity, periodic switching instants, absence of control input), which is motivated by the features of a focused case study: a DC-DC boost converter built by electronics laboratory SATIE (ENS Cachan) for the automotive industry. Our enhanced methods allow us to treat successfully this real-life example.

## 2 Indirect Approach: Approximately Bisimilar Methods

### 2.1 Sampled Switched Systems

In this paper, we consider a subclass of hybrid systems [8], called “switched systems” in [7].

**Definition 1** A switched system  $\Sigma$  is a quadruple  $(\mathbb{R}^n, P, \mathcal{P}, F)$  where:

- $\mathbb{R}^n$  is the state space
- $P = \{1, \dots, m\}$  is a finite set of modes,
- $\mathcal{P}$  is a subset of  $\mathcal{S}(\mathbb{R}_{\geq 0}, P)$  which denotes the set of piecewise constant functions from  $\mathbb{R}_{\geq 0}$  to  $P$ , continuous from the right and with a finite number of discontinuities on every bounded interval of  $\mathbb{R}_{\geq 0}$
- $F = \{f_p \mid p \in P\}$  is a collection of functions indexed by  $P$ .

For all  $p \in P$ , we denote by  $\Sigma_p$  the continuous subsystem of  $\Sigma$  defined by the differential equation:

$$\dot{\mathbf{x}}(t) = f_p(\mathbf{x}(t)).$$

A *switching signal* of  $\Sigma$  is a function  $\mathbf{p} \in \mathcal{P}$ , the discontinuities of  $\mathbf{p}$  are called *switching times*. A piecewise  $\mathcal{C}^1$  function  $\mathbf{x} : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^n$  is said to be a *trajectory* of  $\Sigma$  if it is continuous and there exists a switching signal  $\mathbf{p} \in \mathcal{P}$  such that, at each  $t \in \mathbb{R}_{\geq 0}$ ,  $\mathbf{x}$  is continuously differentiable and satisfies:

$$\dot{\mathbf{x}}(t) = f_{\mathbf{p}(t)}(\mathbf{x}(t)).$$

We will use  $\mathbf{x}(t, x, \mathbf{p})$  to denote the point reached at time  $t \in \mathbb{R}_{\geq 0}$  from the initial condition  $x$  under the switching signal  $\mathbf{p}$ . Let us remark that a trajectory of  $\Sigma_p$  is a trajectory of  $\Sigma$  associated with the constant signal  $\mathbf{x}(t) = p$ , for all  $t \in \mathbb{R}_{\geq 0}$ .

In this paper, we focus on the case of *linear* switched systems: for all  $p \in P$ , the function  $f_p$  is defined by  $f_p(x) = A_p x + b_p$  where  $A_p$  is a  $(n \times n)$ -matrix of constant elements  $(a_{i,j})_p$  and  $b_p$  is a  $n$ -vector of constant elements  $(b_k)_p$ .

In the following, as in [7], we will work with trajectories of duration  $\tau$  for some chosen  $\tau \in \mathbb{R}_{\geq 0}$ , called “time sampling parameter”. This can be seen as a sampling process. Particularly, we suppose that switching instants can only occur at times of the form  $i\tau$  with  $i \in \mathbb{N}$ . In the following, we will consider transition systems that describe trajectories of duration  $\tau$ , for some given *time sampling parameter*  $\tau \in \mathbb{R}_{\geq 0}$ .

**Definition 2** Let  $\Sigma = (\mathbb{R}^n, P, \mathcal{P}, F)$  be a switched system and  $\tau \in \mathbb{R}_{\geq 0}$  a time sampling parameter. The  $\tau$ -sampled transition system associated to  $\Sigma$ , denoted by  $T_\tau(\Sigma)$ , is the transition system  $(Q, \rightarrow_\tau^P)$  defined by:

- the set of states is  $Q = \mathbb{R}^n$

- the transition relation is given by

$$x \rightarrow_{\tau}^p x' \text{ iff } \mathbf{x}(\tau, x, p) = x'$$

Let us define:  $Post_i(X) = \{x' \mid x \rightarrow_{\tau}^i x' \text{ for some } x \in X\}$ , and  $Pre_i(X) = \{x' \mid x' \rightarrow_{\tau}^i x \text{ for some } x \in X\}$ . For the sake of brevity, we will use  $Post_i(x)$  instead of  $Post_i(\{x\})$  and  $Pre_i(x)$  instead of  $Pre_i(\{x\})$ .

**Example 1** This example is a boost DC-DC converter with one switching cell (see Fig. 1) that is taken from [7] (see also, e.g., [3, 5, 11]). The boost converter has two operation modes depending on the position of the switching cell. The state of the system is  $x(t) = [i_l(t), v_c(t)]^T$  where  $i_l(t)$  is the inductor current and  $v_c(t)$  the capacitor voltage. The dynamics associated with both modes are of the form  $\dot{x}(t) = A_p x(t) + b_p$  ( $p = 1, 2$ ) with

$$A_1 = \begin{pmatrix} -\frac{r_l}{x_l} & 0 \\ 0 & -\frac{1}{x_c} \frac{1}{r_0 + r_c} \end{pmatrix} \quad b_1 = \begin{pmatrix} \frac{v_s}{x_l} \\ 0 \end{pmatrix}$$

$$A_2 = \begin{pmatrix} -\frac{1}{x_l} \left( r_l + \frac{r_0 r_c}{r_0 + r_c} \right) & -\frac{1}{x_l} \frac{r_0}{r_0 + r_c} \\ \frac{1}{x_c} \frac{r_0}{r_0 + r_c} & -\frac{1}{x_c} \frac{r_0}{r_0 + r_c} \end{pmatrix} \quad b_2 = \begin{pmatrix} \frac{v_s}{x_l} \\ 0 \end{pmatrix}$$

It is clear that the boost converter is an example of a switched system. We will use the numerical values of [7]:  $x_c = 70$ ,  $x_l = 3$ ,  $r_c = 0.005$ ,  $r_l = 0.05$ ,  $r_0 = 1$ ,  $v_s = 1$ . The goal of the boost converter is to regulate the output voltage across the load  $r_0$ . This control problem is usually reformulated as a current reference scheme. Then, the goal is to keep the inductor current  $i_l(t)$  around a reference value  $i_l^{ref}$ . This can be done, for instance, by synthesizing a controller that keeps the state of the switched system in an invariant set centered around the reference value. An example of switching rule is illustrated on Fig. 2. This rule is periodic of period  $T_d$ : the mode is 2 on  $(0, \frac{T_d}{4}]$  and 1 on  $(\frac{T_d}{4}, T_d]$ . A.

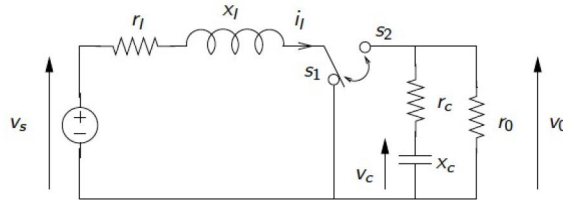


Figure 1: Electric scheme of the boost DC-DC converter (1 cell)

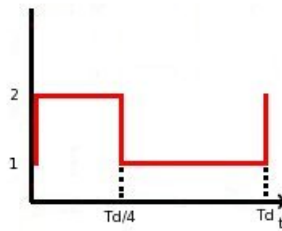


Figure 2: A switching rule for the 1-cell boost DC-DC converter on one period of length  $T_d$  (here,  $\tau = \frac{T_d}{4}$ , and the pattern of the switching rule is (2.1.1.1))

## 2.2 Approximate bisimulation

In [7], the authors propose a method for abstracting a switched system under the form of a discrete symbolic model, that is equivalent to the original one, under certain Lyapunov-based stability conditions. They use an Euclidian metric  $\|\cdot\|$ , and define the approximation of the set of states  $\mathbb{R}^n$  as follows:

$$[\mathbb{R}^n]_\eta = \{x \in \mathbb{R}^n \mid x_i = k_i \frac{2\eta}{\sqrt{n}}, k_i \in \mathbb{Z}, i = 1, \dots, n\},$$

where  $\eta \in \mathbb{R}_{\geq 0}$  is a state space discretization parameter. The transition relation  $T_\tau(\Sigma)$  is approximated as follows: Let  $q \in [\mathbb{R}^n]_\eta$  and  $q_e = \mathbf{x}(\tau, q, p)$  such that  $q \rightarrow_{T_\tau(\Sigma_p)} q_e$  in the real system, let  $q' \in [\mathbb{R}^n]_\eta$  with  $\|q_e - q'\| < \eta$ . Then we have  $q \rightarrow_{T_{\tau,\eta}(\Sigma_p)} q'$  for the approximated transition relation. The *approximate* transition system  $T_{\tau,\eta}(\Sigma)$  is defined as follows:

**Definition 3** The system  $T_{\tau,\eta}(\Sigma)$  is the transition system  $(Q, \rightarrow_{\tau,\eta}^p)$  defined by:

- the set of states is  $Q = [\mathbb{R}^n]_\eta$
- the transition relation is given by

$$q \rightarrow_{\tau,\eta}^p q' \text{ iff } \|\mathbf{x}(\tau, q, p) - q'\| \leq \eta$$

where  $\|\cdot\|$  is any metric on  $\mathbb{R}^n$ .

The notion of “approximate bisimilarity” between systems  $T_\tau(\Sigma)$  and  $T_{\tau,\eta}(\Sigma)$  is defined as follows:

**Definition 4** Systems  $T_\tau(\Sigma)$  and  $T_{\tau,\eta}(\Sigma)$  are  $\varepsilon$ -bisimilar if:

1.  $(\|x - q\| \leq \varepsilon \wedge q \rightarrow_{\tau,\eta}^p q') \Rightarrow \|x' - q'\| \leq \varepsilon$   
for some  $x' = \mathbf{x}(\tau, x, p)$  (i.e. for some  $\mathbf{x}' : x \rightarrow_\tau^p x'$ ), and
2.  $(\|x - q\| \leq \varepsilon \wedge x \rightarrow_\tau^p x') \Rightarrow \|x' - q'\| \leq \varepsilon$   
for some  $q' \in [\mathbb{R}^n]_\eta : \|\mathbf{x}(\tau, q, p) - q'\| \leq \eta$  (i.e. for some  $q' : q \rightarrow_{\tau,\eta}^p q'$ ).

The following theorem is given in [7].

**Theorem 1** Consider a switched system  $\Sigma = (\mathbb{R}^n, P, \mathcal{P}, F)$  with  $\mathcal{P} = \mathcal{S}(\mathbb{R}^+, P)$ , a desired precision  $\varepsilon$  and a time sampling value  $\tau$ . Under certain Lyapunov-based stabilization conditions, there exists a space sampling value  $\eta$  such that the transition systems  $T_\tau(\Sigma)$  and  $T_{\tau,\eta}(\Sigma)$  are approximately bisimilar with precision  $\varepsilon$ .

One can guarantee an arbitrary precision  $\varepsilon$  by choosing an appropriate  $\eta$ : there exists an explicit algebraic relation between  $\varepsilon$  and  $\eta$ . Furthermore, under certain conditions (stability of  $T_\tau(\Sigma)$ ), the symbolic model  $T_{\tau,\eta}(\Sigma)$  has a finite number of states. One can then use standard techniques of model checking in order to synthesize a safe switching rule on  $T_{\tau,\eta}(\Sigma)$  (e.g., letting the system always in the safe area), see e.g. [1, 10]. The switching rule on  $T_{\tau,\eta}(\Sigma)$  can also be used to enforce the real system  $T_\tau(\Sigma)$  to behave correctly.

## 2.3 Simplification for the Case of Linear Dynamics

By focusing on *linear* dynamics, we are allowed to simplify the more general method of [7] as follows:

1. We are using the infinity norm in order to remove the overlapping of two adjacent bowls of radius  $\eta$  (reducing it to a set with a norm 0). This is done to prevent non-determinism. Therefore,  $[\mathbb{R}^n]_\eta$  has to be changed according to the use of this norm. From now on,  $[\mathbb{R}^n]_\eta = \{x \in \mathbb{R}^n \mid x_i = 2k\eta \ k \in \mathbb{Z}\}$

2. The computation of Lyapunov functions are not necessary in our particular case but can be done by simply computing the infinite sum of a geometric serie to ensure the  $\varepsilon$ -bisimulation. Stability criterion relies simply on the eigenvalues of matrices  $A_i$  having negative real part. The proof of  $\varepsilon$ -bisimilarity is based on the fact that  $\beta_\tau \varepsilon + \eta \leq \varepsilon$  (which is true for some  $\eta$  when  $\beta_\tau < 1$ ) (See [6] for more details).
3. Due to the presence of the exponential of a matrix, the computation of the image of all the points could be very costly. By using the linearity of the system, we can compute the same results for a fraction of the initial cost. This is explained in [6].

**Example 2** Our simplified method is applied on the boost converter of Example 1:  $\tau = 0.5$ ,  $V$  corresponding to  $i_l \in [3, 3.4]$  and  $v_c \in [1.5, 1.8]$ ,  $\varepsilon = 3.0$ ,  $\eta = 1/40$ .<sup>1</sup> See Fig. 3 for one of the connected component of the full graph. Each cycle in the graph corresponds to a periodic control of the converter which ensures that the electric variables lie inside the predefined  $V$  up to  $\varepsilon$ . For example, we consider the cycle going through the vertices: 159, 243, 173, 257, 187, 271, 201, 285, 215, 299, 229, 159. This corresponds to the periodic mode control of pattern 12121212122. The result of a simulation under this periodic switching rule is given in Fig. 4 for a starting point  $x_0 = (3.0, 1.79)$ . The box  $V$  is delimited by the red lines. One can see that the system largely exceeds the limits of  $V$  (but stays inside the  $\varepsilon$ -approximation).

### 3 Direct Approach: Inference of Controllable Subspace

The *direct* approach works directly on the continuous state space; this is done, e.g., in [2]. One can thus infer a *controllable* subspace  $V'$  of  $V$ , within which the existence of a switching rule allowing to stay forever within  $V'$  is guaranteed (see, e.g., [12, 9]). We present here a simplified direct method that exploits the simple features of our framework: linearity, absence of perturbation  $u$ , periodicity of the switching instants.

Consider a box  $V \subset \mathbb{R}^n$  and a time sampling value  $\tau$ . The following Algorithm 1 computes a set of controllable polyhedra. Intuitively, after the  $k^{th}$  iteration of the loop,  $Control_i$  is a set of states satisfying the following property: there exists a sequence of modes  $\sigma$  of length  $k$  starting with mode  $i$  such that  $\sigma$  applied to any state of  $Control_i$  prevents the system to go out of  $V$  at any sampling time; alternatively, after the  $k^{th}$  iteration,  $Uncontrol$  is a set of states for which, for all sequence  $\sigma$ , there exists a prefix  $\sigma'$  which makes the system go outside  $V$ . Note that the termination of the procedure is not guaranteed due to the fact that there are infinitely many polyhedral sets.

---

<sup>1</sup>The values used are not the same as the ones used by the authors of [7] due to a rescaling done in [7]

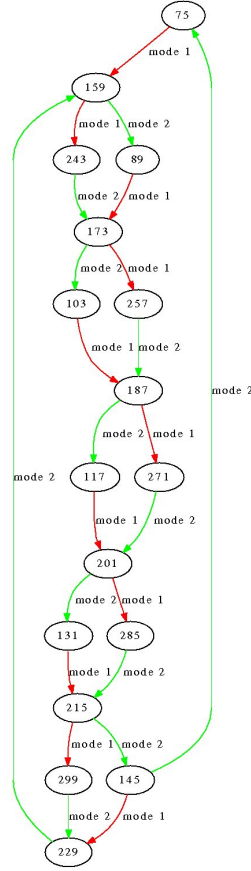


Figure 3: Synthesized finite-state automaton for 1-cell converter with  $\eta = \frac{1}{40}$  and  $V$  corresponding to  $i_l \in [3, 3.4]$  and  $v_c \in [1.5, 1.8]$

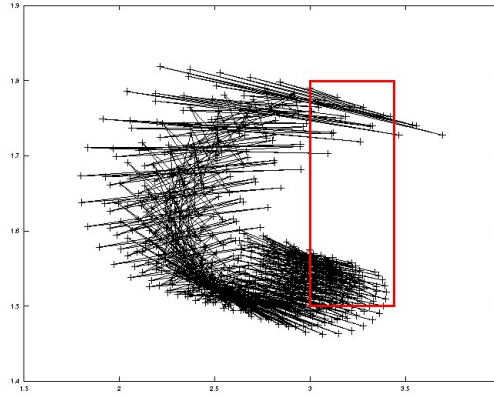


Figure 4: Projected simulation with  $i_l$  on the abscissa, and  $v_c$  on the ordinates of 1-cell converter for switching rule  $(12121212122)^*$  starting point  $x_0 = (3.0, 1.79)$  ; The box  $V = [3, 3.4] \times [1.5, 1.8]$  is drawn in red. ( $\varepsilon = 2.6$ )

**Algorithm 1:** Synthesis of controllable subspace

---

**input** : A switched system  $\Sigma$  with  $m$  modes  
**input** : A time sampling value  $\tau$   
**input** : A box  $V \subset \mathbb{R}^n$   
**output**: A controllable subspace  $V' = \{Control_i\}_{i=1..m}$

```

1  $Uncontrol_{new} := V$ 
2 for  $i = 1..m$  do
3    $Control_i := V$ 
4 repeat
5    $Uncontrol := Uncontrol_{new}$ 
6   for  $i = 1..m$  do
7      $Control_i := Control_i \setminus Pre_i(Uncontrol)$ 
8    $Uncontrol_{new} := V \setminus \bigcup_{i=1}^m Control_i$ 
9 until  $Uncontrol_{new} = Uncontrol$ ;
10 return  $V' = \bigcup_{i=1}^m Control_i$ 

```

---

The correctness of Algorithm 1 relies on the following fact:

**Theorem 2** *If Algorithm 1 terminates with output  $V'$ , then  $Post_i(V') \subseteq V'$  for all  $1 \leq i \leq m$ .*

In order to prove Theorem 2, we need the two following propositions.

**Proposition 1** *Let  $x, x' \in \mathbb{R}^n$ . Then the three following items are equivalent:*

- $x = x'$
- $Post_i(x) = Post_i(x')$  for all  $1 \leq i \leq m$
- $Pre_i(x) \neq Pre_i(x')$  for all  $1 \leq i \leq m$

**Proof**

We only consider Linear Differential Equations, for any  $x \in \mathbb{R}^n$   $Card(Post_i(x)) = Card(Pre_i(x)) = 1$ . The proof is immediate from Cauchy-Lipschitz Theorem.  $\square$

**Proposition 2** *Let  $A, B \subseteq \mathbb{R}^n$ . We have:*

1.  $Post_i(A \setminus B) = Post_i(A) \setminus Post_i(B)$  and similary for  $Pre_i$  for all  $1 \leq i \leq m$ .
2.  $Post_i(Pre_i(A)) = Pre_i(Post_i(A)) = A$ .

**Proof**

Let  $A, B \subseteq \mathbb{R}^n$  and  $i \in \{1..m\}$ . We have:

1. Let  $y \in Post_i(A \setminus B)$ . Then  $x \rightarrow_{\tau}^i y$  for some  $x \in A \setminus B$ . Therefore  $y \in Post_i(A)$  since  $x \in A$ . Let suppose that  $y \in Post_i(B)$  then  $x' \rightarrow_{\tau}^i y$  for some  $x' \in B$ . By Proposition 1,  $x = x'$  which raises a contradiction. Therefore  $y \notin Post_i(B)$  and  $y \in Post_i(A) \setminus Post_i(B)$ .  
 Let  $y \in Post_i(A) \setminus Post_i(B)$ . Therefore,  $y \in Post_i(A)$  and  $x \rightarrow_{\tau} y$  for some  $x \in A$ . However,  $y \notin Post_i(B)$ , therefore  $\forall x' \in B, x' \not\rightarrow_{\tau} y$ . It follows that  $x \in A \setminus B$ . Therefore  $y \in Post_i(A \setminus B)$   $\square$
2. We know that  $Card(Post_i(Pre_i(x))) = 1$  and it is immediate that  $x \in Post_i(Pre_i(x))$  therefore  $\{x\} = Post_i(Pre_i(x))$ . Therefore  $Post_i(Pre_i(A)) = A$  and by the same type of proof  $Pre_i(Post_i(A)) = A$ .  $\square$

We can now prove Theorem 2

### Proof of Theorem 2

We will denote by  $Control_i^k$  the value of  $Control_i$  at the  $k^{th}$  iteration of the algorithm. Suppose that the Algorithm terminates at iteration  $n$ . Therefore, at iteration  $n$ ,  $Uncontrol_{new} = Uncontrol$ . Let  $x \in Control_i^n$  for a given  $i \in \{1..m\}$ . Suppose that  $Post_i(x) \notin V'$  then  $Post_i(x) \in Uncontrol_{new}$ . Therefore  $Post_i(x) \in Uncontrol$ . By Proposition 2,  $x \in Pre_i(Uncontrol)$ , but we have that  $Control_i^n = Control_i^{n-1} \setminus Pre_i(Uncontrol)$ . Therefore  $x$  does not belong to  $Control_i^n$  which raises a contradiction. Therefore  $Post_i(x) \in V'$  and  $Post_i(V') \subseteq V'$   $\square$

In other words, the output set  $V'$  of controllable polyhedra is invariant. Let us point out that the system may temporarily go out of  $V$  between two sampling instants.

It immediately follows from Theorem 2 that, at any sampling time  $i\tau$ , any point in  $V' \subset V$ , is controllable: there exists a mode  $j$  that ensures that the point at next sampling time  $(i+1)\tau$  is still in  $V'$ . Note that, unlike the indirect method, the appropriate mode cannot be precomputed, but has to be found *on line*. On the other hand, the system lies *exactly* within  $V' \subset V$  at each sampling time (instead of lying within the  $\varepsilon$ -closeness of  $V$  using the indirect method).

Algorithm 1 involves the computation of the *Pre*-image of (union of) convex polyhedra. We have (see [6] for a proof):

**Lemma 1** *Let  $\tau \in \mathbb{R}_{>0}$  and  $S$  a convex set of  $\mathbb{R}^n$  and  $i$  a mode of  $\Sigma$ . Then  $Pre_i(S)$  is a convex set of  $\mathbb{R}^n$ .*

From this Lemma, we can compute the *Pre*-image of any convex polyhedron by simply computing the *Pre*-image of its vertices.

Unfortunately, Algorithm 1 also involves the computation of union, complementation, and test of equality of polyhedra, that are operations known to be very expensive. To overcome this problem, one can approximate all the manipulated objects using the notion of *griddy polyhedra* (see [2, 4]), i.e., sets that can be written as unions of closed unit hypercubes with integer vertices. The price to be paid is an underapproximation of the controllability subspace, but this kind of compromise seems unavoidable, as pointed out in [2].

**Example 3** *To illustrate this approach, we are computing a control for the boost DC-DC converter with one cell, see Example 1 for a description of the system. The resulting control presented in Fig. 6 has been obtained for the following parameters:  $V$  corresponding to  $i_l \in [3.0, 3.4]$  and  $v_c \in [1.5, 1.8]$ ,  $\tau = 0.5$  and  $x_0 = (3.01, 1.79)$ . The Fig. 6 can be decomposed into 4 parts:*

- *Two big vertical polyhedra. The left one represents the zone controllable with mode 1, the right one with mode 2.*
- *Two small horizontal polyhedra (upper right and lower left) are the uncontrollable zones of  $V$ .<sup>2</sup>*

*A trajectory starting from point  $x_0 = (3.01, 1.79)$  belonging to the controllable subset, and using an on-line computation of the switching rule, has been depicted on Fig. 6. and 5 for a simulation. One can see that the trajectory always stays within  $V$  (not only at the sampling instants).*

## 4 Application to a 3-cells DC-DC Boost Converter

Our method is scalable to bigger systems as we illustrate with the boost DC-DC converter with 3 cells. This is a real-life system built by the electronics laboratory SATIE (ENS Cachan) for the automotive industry. See Fig. 7 for a picture of the system.

<sup>2</sup>They are delimited by vertices:  $(3, 1.5); (3.1092, 1.5); (3.1110, 1.5107); (3.0000, 1.5107)$  and  $(3.2611, 1.7897); (3.4, 1.788); (3.4, 1.8); (3.2611, 1.8)$ .



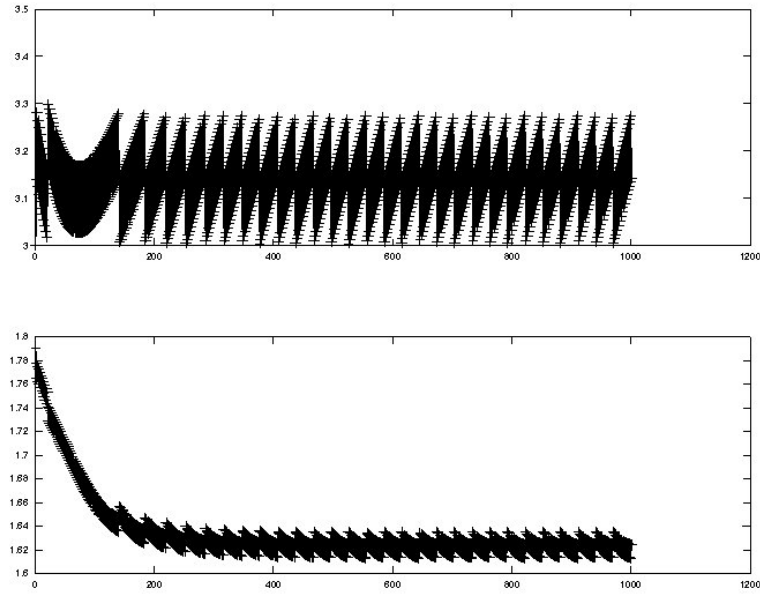


Figure 5: Simulation of the 1-cell boost converter starting from point  $x_0 = (3.01, 1.79)$  that belongs to the controllable subspace of  $[3.0, 3.4] \times [1.5, 1.8]$ . Above: simulation of  $v_c$  during time; below: simulation of  $i_l$

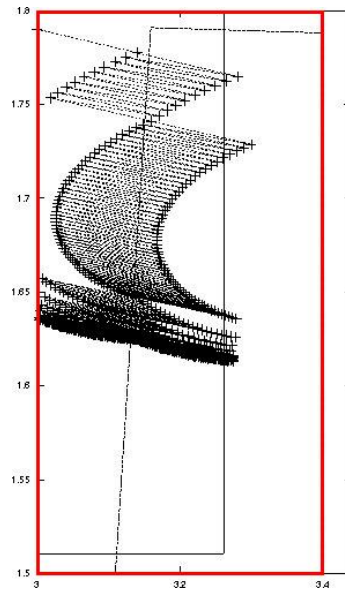


Figure 6: Controllable subspaces of the boost DC-DC converter (1 cell) from  $x_0 = (3.01, 1.79)$  within  $V = [3.0, 3.4] \times [1.5, 1.8]$

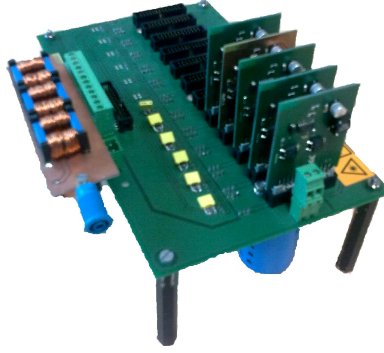


Figure 7: 3-cells converter built by laboratory SATIE

#### 4.1 Model

The boost DC-DC converter with 3 cells relies on the same principle as the one with one cell. An advantage of this model is its robustness: even if one switching cell is damaged, the system is still controllable with the restricted set of modes that remain available. This system is naturally more complex: There are 4 continuous variables of interest (instead of two), and  $2^3 = 8$  modes (instead of two). The electrical scheme is presented in Fig. 8. An example of pattern of periodic switching rule is presented in Fig. 9.

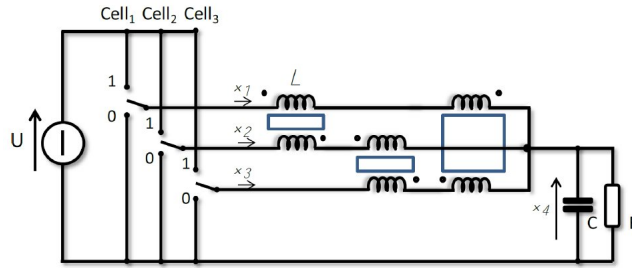


Figure 8: Electrical scheme of the DC-DC converter with 3 cells

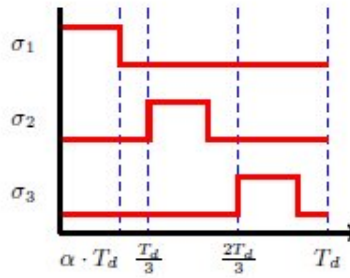


Figure 9: Switching rule for the 3-cells boost DC-DC converter on one period of length  $T_d$  (here,  $\alpha = \frac{1}{2}$ ,  $\tau = \frac{T_d}{6}$ ,  $\sigma_1 = (1.0^5)$ ,  $\sigma_2 = (0^2.1.0^3)$ ,  $\sigma_3 = (0^4.1.0)$ , and the pattern of the corresponding switching rule is  $(2.1.3.1.5.1)$ )

The system satisfies the following equations:

That can be rewritten to fit our framework as:

with

where  $U$  is the input voltage (here  $U = 100$ ).

Here are the parameters that we used:  $\eta = 1/5$  which corresponds to  $\varepsilon = 21.6$ ,  $\alpha = \frac{1}{2}$ ,  $T_d = 1/10000$ ,  $\tau = 1/60000$ .  $V$  is defined by:  $[5.3, 5.9] \times [5.3, 5.9] \times [5.3, 5.9] \times [15.5, 16.5]$ .

corresponds to the pattern modes 444121. The construction of the full graph (including some optimizations, like the deletion of vertices from which every rule leads to a deadend) took less than 2 minutes.

From this graph, we extracted several cycles that correspond to different switching rules. We have simulated the system starting from point  $x_0 = (5.4, 5.4, 5.4, 16)$  for such various rules. The result of one simulation for one of them (viz.,  $(444121)^*$ ) is given in Fig. 11. We can see that, under all these

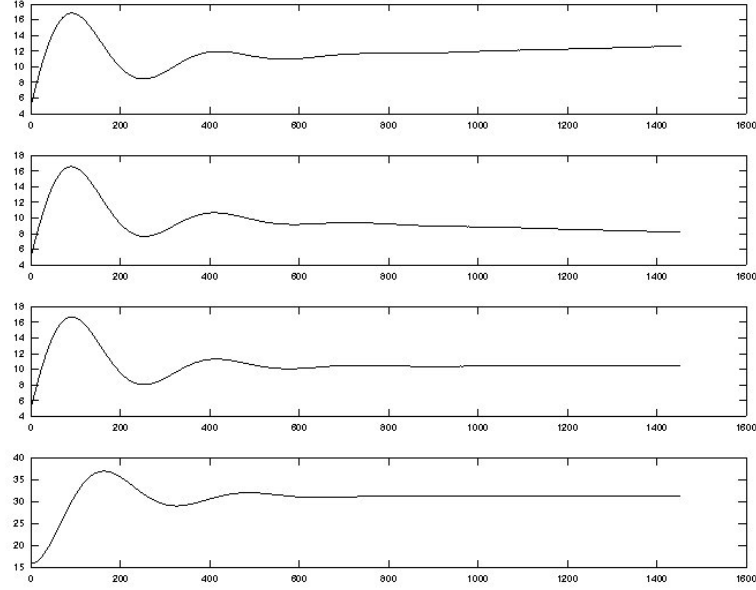


Figure 11: Simulation of the 3-cells Converter under Switching Rule  $(333010)^*$

controls, the system goes out of the initial  $V$ . However we can check that the system stays within the  $\varepsilon$ -overapproximation of  $V$  with  $\varepsilon = 21.6$ . Let us point out incidentally that such an  $\varepsilon$  is much too gross to guarantee a realistic precision. Rather than presenting the results of the indirect approach with a better precision (using a finer  $\eta$ -grid), we present henceforth the results obtained with the direct approach.

### 4.3 Direct Method

For  $V = [4, 7] \times [4, 7] \times [4, 7] \times [15, 17]$ ,  $\tau = 1/60000$ , we can extract a controllable subspace  $V' \subset V$ . A simulation of the system starting from  $x_0 = (5, 5, 5, 16) \in V'$  is presented in Fig. 12 (see also Fig. 13 for a projected simulation). We can check on the figure that all the simulation lies within  $V$ .

### 4.4 Control on failure

We have also experimented the methods in a case of failure of one switching cell of the 3-cells boost converter: we have supposed that the cell 1 is stuck on position open ( $\sigma_1 = 0$ ), which means that only 4 of the 8 modes are still available. The description of our experiments is beyond the scope of this paper. Let us just point out that we were able to find a switching rule for this downgraded context using the *direct* method, but not with the indirect approach.

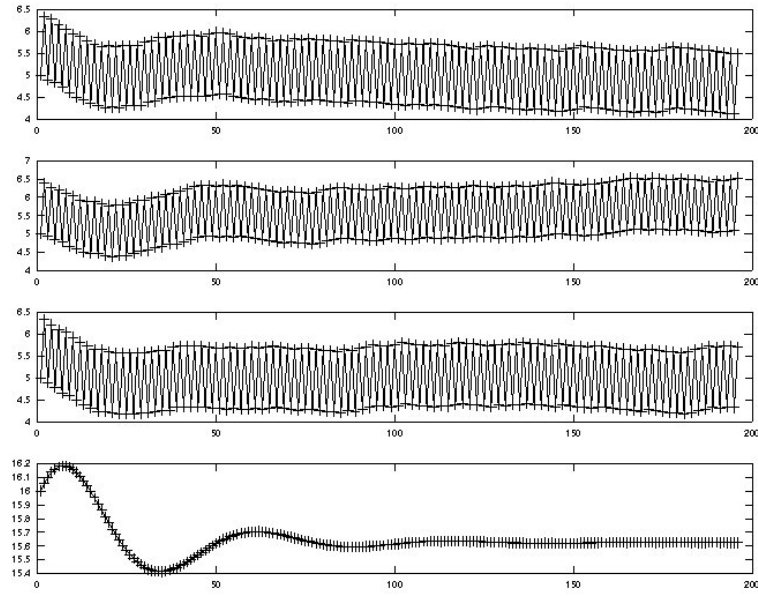


Figure 12: Simulation of 3-cells converter starting from  $x_0 = (5, 5, 5, 16)$  in  $V = [4, 7] \times [4, 7] \times [4, 7] \times [15, 17]$  (from top to bottom:  $x_1, x_2, x_3, x_4$  in function of time)

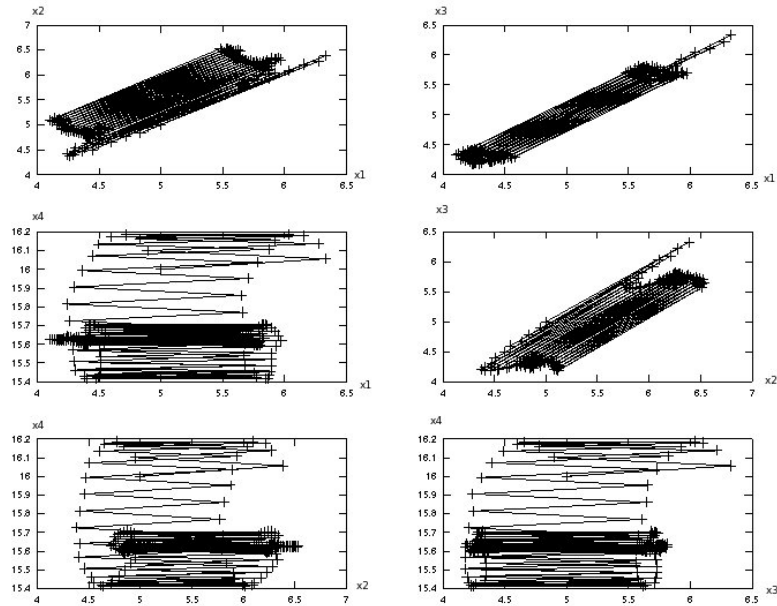


Figure 13: Various projections on plans  $(x_i, x_j)$  (for  $i, j \in \{1, \dots, 4\}$ ) of simulations of the 3-cells boost converter starting from  $x_0 = (5, 5, 5, 16)$  in  $V = [4, 7] \times [4, 7] \times [4, 7] \times [15, 17]$

## 5 Final Remarks

We have explained how to improve two methods (the direct and indirect ones) for synthesizing control of a piecewise linear system by exploiting the special features of a framework met in the case of a real-case example. Our experiments show that the advantage of the indirect method is to allow the user to precompute a periodic control rule at the price of a certain loss of precision. On the other hand, the direct method relies on a on-line computation of the switching rule, but allows us to satisfy *exact* reachability invariance properties. Furthermore, the direct method seems to be able to treat more easily limit cases where the system works in a downgraded configuration due to a failure of one its components.

## Acknowledgment

We are grateful to Laurent Doyen for his helpful comments on an earlier draft of this paper.

## References

- [1] A. Arnold, A. Vincent & I. Walukiewicz (2003): *Games for synthesis of controllers with partial observation*. *Theor. Comput. Sci.* 303, pp. 7–34, doi:10.1016/S0304-3975(02)00442-5.
- [2] E Asarin, O Bournez, T Dang, O Maler & A Pnueli (2000): *Effective Synthesis of Switching Controllers for Linear Systems*. *Proceedings of the IEEE, Special Issue on Hybrid Systems* 88, pp. 1011–1025, doi:10.1109/5.871306.
- [3] A.G. Beccuti, G. Papafotiou & M. Morari (2005): *Optimal Control of the Boost dc-dc Converter*. In: *IEEE Conference on Decision and Control*, Seville, Spain. pp 4457–4462, doi:10.1109/CDC.2005.1582864.
- [4] O. Bournez, O. Maler & A. Pnueli (1999): *Orthogonal polyhedra: Representation and computation*. In: *HSCC, LNCS 1569*, Springer, pp. 46–60, doi:10.1007/3-540-48983-5\_8.
- [5] J. Buisson, P.-Y. Richard & H. Cormerais (2005): *On the Stabilisation of Switching Electrical Power Converters*. In: *HSCC, LNCS 3414*, pp. 184–197, doi:10.1007/978-3-540-31954-2\_12.
- [6] L. Fribourg, B. Revol & R. Soulat (2011): *Synthesis of Switching Rules for Ensuring Reachability Properties of Sampled Linear Systems*. Research Report LSV-11-12, Laboratoire Spécification et Vérification, ENS Cachan, France.
- [7] A. Girard, G. Pola & P. Tabuada (2010): *Approximately Bisimilar Symbolic Models for Incrementally Stable Switched Systems*. *IEEE Trans. on Automatic Control* 55, pp. 116–126, doi:10.1109/TAC.2009.2034922.
- [8] T. A. Henzinger (1996): *The theory of hybrid automata*. In: *Proceedings of the 11th Annual IEEE Symposium on Logic in Computer Science, LICS '96*, IEEE Computer Society, Washington, DC, USA, pp. 278–292, doi:10.1109/LICS.1996.561342.
- [9] M. I. Krastanov & V. M. Veliov (2005): *On the controllability of switching linear systems*. *Automatica* 41(4), pp. 663 – 668, doi:10.1109/TAC.2002.801182.
- [10] P. J. G. Ramadge & W. M. Wonham (1989): *The control of discrete event systems*. *Proceedings of The IEEE* 77, pp. 81–98, doi:10.1109/5.21072.
- [11] M. Senesky, G. Eirea & T. John Koo (2003): *Hybrid modeling and control of power electronics*. In: *HSCC, LNCS 2623*, Springer, pp. 450–465, doi:10.1007/3-540-36580-X\_33.
- [12] Z. Sun, S. S. Ge & T. H. Lee (2002): *Controllability and reachability criteria for switched linear systems*. *Automatica* 38(5), pp. 775–786, doi:10.1016/S0005-1098(01)00267-9.